

Efficient Quantum Algorithms for Shifted Quadratic Character Problems

Wim van Dam
UC Berkeley, CWI Amsterdam
vandam@cs.berkeley.edu

Sean Hallgren
MSRI
hallgren@cs.berkeley.edu

February 1, 2008

Abstract

We introduce the Shifted Legendre Symbol Problem and some variants along with efficient quantum algorithms to solve them. The problems and their algorithms are different from previous work on quantum computation in that they do not appear to fit into the framework of the Hidden Subgroup Problem. The classical complexity of the problem is unknown, despite the various results on the irregularity of Legendre sequences.

1 Introduction

All known problems that have a polynomial time quantum algorithms but have no known polynomial time classical algorithm are some variant of the Hidden Subgroup Problem. The problem is: given a function on a group G that is constant and distinct on cosets of some unknown subgroup, find a set of generators of the subgroup. An example of a problem that can be viewed in this framework is Shor's algorithm for factoring. In this case the problem reduces to finding the period of a function, which amounts to finding the hidden subgroup of a cyclic group. The variant in this case is that the group size is unknown. Much of the research in quantum algorithms has focused on first reducing the problem to a variant of the Hidden Subgroup Problem (HSP), and then extending the machinery to handle the particular variant. Some examples of this include [5, 10, 11, 12, 13, 14, 16, 18, 19].

In this paper we introduce the Shifted Legendre Symbol Problem (SLSP), which does not appear to be an instance of the Hidden Subgroup Problem. As such, the quantum algorithm for the SLSP deviates from the structure of the algorithms that solve the HSP. The quantum component of the previous algorithms has two basic steps. The first is to set up an equal superposition over group elements by computing the Fourier transform, after which some function evaluation is executed, such as $|x\rangle \rightarrow |x, f(x)\rangle$ or $|x\rangle \rightarrow (-1)^{f(x)}|x\rangle$. The second step is to Fourier sample the state [4], i.e. to compute the Fourier transform and measure. The algorithm in this article starts with the state setup as before, but Fourier sampling is not sufficient, because after the second step the resulting distribution is uniform no matter what instance of the problem is given. This is very different from the HSP algorithm, where the distribution induced by Fourier sampling is enough to solve the problem. There is also another difference. In general, a Fourier transform is defined in

terms of a group, and as a result previous algorithms relied solely on properties of groups, such as what their irreducible representations are. One variant of the SLSP we solve is its extension to general finite fields. In contrast to previous algorithms, our algorithm uses a transform that depends on the fact that there is an underlying field, which supports both addition and multiplication.

The Shifted Legendre Symbol Problem is defined as follows. Given a function f and an odd prime p such that $f(x) = \left(\frac{x+s}{p}\right)$, find s . Here $\left(\frac{x}{p}\right)$ is the Legendre Symbol, which is 1 if x is a square mod p , -1 if it is not, and 0 if $p|x$. We also introduce a few variants of this problem. The first variation is the Shifted Jacobi Symbol Problem. The setup is the same as the original problem, except that instead of a prime p , an odd square free $n = p_1 \cdots p_k$ is used and the function is $f_s(x) = \left(\frac{x+s}{n}\right)$, where $\left(\frac{x}{n}\right)$ is the Jacobi symbol. The second variant also keeps n unknown. To still be able to define the function a new domain of size M is used where M is an arbitrary integer much larger than n and the shifted Jacobi symbol is repeated out to M . The quantum algorithm uses a property of the Jacobi symbol to first find the period n of the function, and then uses the algorithm for the Shifted Jacobi Symbol Problem. The last variant is a generalization of the SLSP to general fields. Here the function is a shifted version of the quadratic character χ over the finite field: $\chi(x)$ is 1 if x is a square in \mathbb{F}_q , -1 if it is not a square, and $\chi(0) = 0$. The algorithm in this case is more complicated and involves properties not existing in purely group theoretic problems. In particular the trace of field elements is used in the transform.

The classical complexities of the problems are unknown. When the field size is known, the problem has polynomial query complexity, but it is unknown if they have polynomial time algorithms. In any case, we hope that the general structure of our algorithm will lead to new quantum algorithms.

Related Work. Finding an efficient quantum algorithm for the Shifted Legendre Symbol Problem was originally posed as an open question in van Dam [8].

Many papers have studied the properties of Legendre and Jacobi sequences, as referenced in [3, 9, 17]. In cryptography, Damgård [9] has suggested using shifted Legendre and Jacobi sequences as pseudo-random bits in the following sense. The seed to the generators are two unknown values s and p . Consider the sequence $\left(\frac{s}{p}\right), \left(\frac{s+1}{p}\right), \dots, \left(\frac{s+t-1}{p}\right)$, where t is a polynomial in $\log p$. If it is hard to predict the next bits $\left(\frac{s+t}{p}\right), \left(\frac{s+t+1}{p}\right), \dots$ from this sequence, then we consider the Legendre sequence ‘unpredictable’. Damgård showed that if Legendre sequences are unpredictable in a very weak sense, then Jacobi sequences (defined similarly) are unpredictable in a very strong sense. Whether or not Legendre sequences are indeed unpredictable is still an open question.

The SLSP with unknown p is at least as hard as the problem posed by Damgård in the sense that solving the SLSP yields the value s , with which the next bits can be computed. However, it is also potentially easier to solve because an SLSP algorithm is allowed to query the string adaptively.

Current problems that have exponential separations between their quantum and classical complexity can be viewed as variants of the Hidden Subgroup Problem. The algorithm to solve these problems first creates a state that is uniform over a coset of a subgroup and then computes the Fourier transform and measures. The recursive Fourier sampling problem [4] is not directly an instance, but uses the same exact properties in reverse. That is, in the Hidden Subgroup Problem algorithm the Fourier transform takes a state that is uniform over a coset to a perp subgroup state where the coset is encoded in the phases of the basis vectors. The reverse operation is to

start with the phases, and to compute the Fourier transform to find the coset. One level of the recursive Fourier sampling problem does this with the subgroup restricted to trivial: the map is $\sum_{x \in \mathbb{Z}_2^n} (-1)^{x \cdot s} |x\rangle \longrightarrow |s\rangle$, where s is the coset. This problem is perhaps the closest in structure to ours, but as mentioned, the new algorithms presented here do not rely only on this property of cosets being taken to perp subgroups with phases.

2 Preliminaries

In this section we first define the problems we solve and then we provide other background necessary for the rest of the paper.

For a prime p , the *Legendre Symbol* $\left(\frac{x}{p}\right)$ is defined to be 1 if x is a quadratic residue, -1 if x is a quadratic non-residue modulo p , and 0 if $p|x$. The Legendre symbol can be extended in several ways. Here we will do so by defining it for rings \mathbb{Z}_n and finite fields \mathbb{F}_q . For an integer $n = p_1 \cdots p_k$ the *Jacobi Symbol* $\left(\frac{x}{n}\right)$ is defined by $\left(\frac{x}{n}\right) = \left(\frac{x}{p_1}\right) \cdots \left(\frac{x}{p_k}\right)$, where the respective $\left(\frac{x}{p_i}\right)$ are Legendre Symbols and the product is over all the prime factors p_i of n , with repetitions. For a finite field \mathbb{F}_q and $x \in \mathbb{F}_q$, the *quadratic character* $\chi(x)$ is 1 if x is a quadratic residue, -1 if x is a quadratic non-residue, and 0 if $x = 0$.

We can now define the problems solved in this paper. The first problem is the basic example which the others build on.

Definition 1 (Shifted Legendre Symbol Problem) *Given an odd prime p and a function $f_s : \mathbb{F}_p \rightarrow \{-1, 0, 1\}$ such that $f_s(x) = \left(\frac{x+s}{p}\right)$ for some $s \in \mathbb{F}_p$, find s .*

The first variant extends the definition to rings.

Definition 2 (Shifted Jacobi Symbol Problem) *Given a square free odd integer n and a function $f_s : \mathbb{Z}_n \rightarrow \{-1, 0, 1\}$ such that $f_s(x) = \left(\frac{x+s}{n}\right)$. Find the unknown shift factor $s \in \mathbb{Z}_n$.*

If the integer n is not square free, the Shifted Jacobi Problem does not have a unique answer anymore. Consider for example the equality, $\left(\frac{x+p}{p^2}\right) = \left(\frac{x+p}{p}\right) \left(\frac{x+p}{p}\right) = \left(\frac{x}{p}\right) \left(\frac{x}{p}\right) = \left(\frac{x}{p^2}\right)$, for all $x \in \mathbb{Z}_{p^2}$. Instead we could define the task to find one of the values s' such that $f_s(x) = \left(\frac{x+s'}{n}\right)$. This problem is again efficiently solvable on a quantum computer.

The goal of the second variant is to also keep n unknown in the Shifted Jacobi Symbol Problem. Notice that the SLSP with p unknown is a special case of this problem.

Definition 3 (Shifted Jacobi Symbol Problem, unknown n) *Given an integer M and a function $f_s : \mathbb{Z}_M \rightarrow \{-1, 0, 1\}$ such that $f_s(x) = \left(\frac{x+s}{n}\right)$ for some integer odd square free n , with $n^2 < M$, find s and n .*

The last variant is a generalization to all possible finite fields.

Definition 4 (Shifted Quadratic Character Problem) *Given $q = p^r$, a power of an odd prime p , and a function $f : \mathbb{F}_q \rightarrow \{-1, 0, 1\}$ such that $f(x) = \chi(x+s)$ for some $s \in \mathbb{F}_q$, find s . Here χ is the quadratic character of \mathbb{F}_q .*

We will now give some background on finite fields, the representation we use, and the time it takes to do basic computations.

Consider the finite field \mathbb{F}_q with q the r th power of the prime p . In this article the elements $x \in \mathbb{F}_q$ are represented as polynomials in $\mathbb{F}_p[X]$ modulo an irreducible polynomial in $\mathbb{F}_p[X]$ of degree r . The \mathbb{F}_p coefficients of such a polynomial are denoted by x_j , that is $x = \sum_{j=0}^{r-1} x_j X^j$. When we write $|x\rangle$ we mean $|x_0, x_1, \dots, x_{r-1}\rangle$. With this representation the bit-complexity of adding or subtracting two elements of \mathbb{F}_q is $O(\log q)$. Multiplication and division require $O((\log q)^2)$ bit operations for this ‘model’ for \mathbb{F}_q . (See, for example, Chapter 6 in [1] for details on this.)

For a finite field \mathbb{F}_{p^r} the *trace* of an element x is defined by

$$\text{Tr}(x) = \sum_{j=0}^{r-1} x^{p^j}.$$

By the equality $(\text{Tr}(x))^p = \text{Tr}(x)$ we see that the trace maps the elements of the finite field to its base field \mathbb{F}_p . Because $\text{Tr}(x)$ is a polynomial of degree p^{r-1} (less than p^r), it is a non-constant function. We also have, using $(x + y)^p = x^p + y^p$,

$$\text{Tr}(ax + by) = a\text{Tr}(x) + b\text{Tr}(y)$$

for all $a, b \in \mathbb{F}_p$ and $x, y \in \mathbb{F}_q$. It follows that the trace of $x = \sum_{j=0}^{r-1} x_j X^j$ equals the summation $\sum_{j=0}^{r-1} x_j \text{Tr}(X^j)$ over the base field \mathbb{F}_p . With this property it can be shown that the calculation of $\text{Tr}(x)$ requires $O((\log q)^2)$ bit operations [1].

The main operation used in quantum algorithms is the Fourier transform. Here we will need to compute the Fourier transform over \mathbb{Z}_p for a large prime p , which is defined by

$$|x\rangle \longrightarrow \frac{1}{\sqrt{p}} \sum_{y=0}^{p-1} e^{2\pi i(xy)/p} |y\rangle.$$

It is unknown how to efficiently compute this transformation exactly. Approximations have been given in [12, 14]. From [12] we have: there is a quantum algorithm which ϵ -approximates the quantum Fourier transform over \mathbb{Z}_p for an arbitrary n -bit p and any ϵ and which runs in time $O(n \log \frac{n}{\epsilon} + \log^2 \frac{1}{\epsilon})$. We will denote the p th root of unity $e^{2\pi i/p}$ by ω_p .

We will also need a result about Fourier sampling (computing the Fourier transform and measuring) repeated superpositions [12]. Suppose we want to compare the distribution induced by Fourier sampling a state $|\phi\rangle$ with the distribution induced by Fourier sampling the state $|\tilde{\phi}\rangle$ which is $|\phi\rangle$ repeated many times. It turns out this is possible but we need to define special distributions to do it. The problem is that $|\tilde{\phi}\rangle$ has a larger support, so we need a way to shrink the domain so the two can be compared. The way to do it is to use continued fractions on the result of the sample. We will now formalize this.

For simplicity we will suppress a detail or two to make this more readable. Let $|\phi\rangle = \sum_{x=0}^{n-1} \phi_x |x\rangle$ be an arbitrary superposition, and let $\hat{\mathcal{D}}_{|\phi\rangle}$ the distribution induced by Fourier sampling $|\phi\rangle$. Let the superposition $|\tilde{\phi}\rangle = c \cdot \sum_{x=0}^{M-1} \phi_{x \bmod n} |x\rangle$ be $|\phi\rangle$ repeated until some arbitrary integer M , not necessarily a multiple of n , where c is the proper normalization constant. Let $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}$ be the distribution

induced by Fourier sampling $|\tilde{\phi}\rangle$. Notice that $\hat{\mathcal{D}}_{|\phi\rangle}$ is a distribution on $\{0, \dots, n-1\}$ and $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}$ is a distribution on $\{0, \dots, M-1\}$.

We can now define the two distributions we will compare. Let $\hat{\mathcal{D}}_{|\phi\rangle}^{\text{RF}}$ be the distribution induced on the reduced fractions of $\hat{\mathcal{D}}_{|\phi\rangle}$, that is, if x is a sample from $\hat{\mathcal{D}}_{|\phi\rangle}$, we will return the fraction x/n in lowest terms. In particular, define $\hat{\mathcal{D}}_{|\phi\rangle}^{\text{RF}}(j, k) = \hat{\mathcal{D}}_{|\phi\rangle}(jm)$ if $mk = n$. Let $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}^{\text{CF}}$ be the distribution induced on fractions from first sampling $\hat{\mathcal{D}}_{|\tilde{\phi}\rangle}$ and then running continued fractions on the result and M . If $M = \Omega(\frac{n}{\epsilon^2})$ and $M = \Omega(\frac{M}{\epsilon})$, then $|\hat{\mathcal{D}}_{|\phi\rangle}^{\text{RF}} - \hat{\mathcal{D}}_{|\tilde{\phi}\rangle}^{\text{CF}}|_1$ is upper bounded by about n/\sqrt{M} .

This basically says that to understand the distribution induced by Fourier sampling a repeated state, only the distribution induced by Fourier sampling non-repeated state has to be understood. However, it is not the exact distribution of the unrepeated state, since we look at the distribution over reduced fractions. We will use this to solve the unknown n case of the Jacobi problem.

3 An Algorithm for Prime Size Fields

In this section we give algorithms solving the Shifted Legendre Symbol Problem and variants when working over a finite field of prime size. The main ideas are contained in the algorithm for the Shifted Legendre Symbol Problem, and we can apply the same algorithm to solve the same problem when p is unknown, and also to solve the Shifted Jacobi Symbol Problem.

The idea for the algorithm follows from a few known facts. Assume we start the algorithm in the standard way, i.e. by putting the function value in the phase to get $|f_s\rangle = \sum_{i \in \mathbb{Z}_p} \left(\frac{i+s}{p}\right) |i\rangle$. Assume the functions f_i are orthogonal (they are close to orthogonal). Define the matrix C where the i^{th} row is $|f_i\rangle$. Our quantum state $|f_s\rangle$ is one of the rows, so $C|f_s\rangle = |s\rangle$. The issue now is how to efficiently implement C . C is a circulant matrix, i.e. $c_{i,j} = c_{i+1,j+1}$. The Fourier transform diagonalizes a circulant matrix: $C = F_p(F_p^{-1}CF_p)F_p^{-1} = F_pDF_p^{-1}$, where D is diagonal, so we can implement C if we can implement D . It turns out that the vector on the diagonal of D is the vector $F_p|f_0\rangle$, but $|f_0\rangle$ is an eigenvector of the Fourier transform, so up to a global phase which we can ignore, we are done. To summarize: to implement C , we compute the Fourier transform, compute f_0 into the phases (this is just the Legendre Symbol), and then compute the Fourier transform again (it is not important whether we use F_p or F_p^{-1}). We will now present this algorithm step-by-step.

Algorithm 1 (Shifted Legendre Symbol Problem)

Input: An odd prime p and a function f_s such that $f_s(x) = \left(\frac{x+s}{p}\right)$ for all $x \in \mathbb{Z}_p$.

Output: s .

1. Compute the Fourier transform over \mathbb{Z}_p of $|0\rangle$ and compute f_s into the phases, approximating:

$$\frac{1}{\sqrt{p-1}} \sum_{x \in \mathbb{F}_p} \left(\frac{x+s}{p}\right) |x\rangle$$

2. Compute the Fourier transform over \mathbb{Z}_p :

$$\frac{1}{\sqrt{p-1}} \sum_{y \in \mathbb{F}_p} \omega_p^{-ys} \left(\frac{y}{p}\right) |y\rangle$$

3. Compute f_0 into the phases, approximating:

$$\frac{1}{\sqrt{p}} \sum_{y \in \mathbb{F}_p} \omega_p^{-ys} |y\rangle$$

4. Compute the inverse Fourier transform over \mathbb{Z}_p ; this gives the answer $|-s\rangle$.

Theorem 1 *Algorithm 1 solves the Shifted Legendre Symbol Problem in two queries and polynomial time with probability exponentially close to one.*

Proof: The first step is a standard setup used in quantum algorithms. The only difference is that f_s evaluates to zero in one position. In this case, just treat it as a one. After this the state is exponentially close to the state shown. Recall that the Legendre Symbol $\left(\frac{x}{p}\right)$ is zero when $p|x$, so one amplitude is zero.

The result of applying the Fourier transform is (where we replace x with $x - s$)

$$\frac{1}{\sqrt{p-1}} \sum_{x=0}^{p-1} \left(\frac{x+s}{p}\right) |x\rangle \longrightarrow \frac{1}{\sqrt{p-1}} \sum_{y=0}^{p-1} \frac{1}{\sqrt{p}} \sum_{x=0}^{p-1} \left(\frac{x}{p}\right) \omega_p^{y(x-s)} |y\rangle.$$

Factoring out the ω_p^{-ys} term, using the change of variable $z = xy$, and using the facts that $\left(\frac{zy^{-1}}{p}\right) = \left(\frac{z}{p}\right) \left(\frac{y^{-1}}{p}\right)$ and $\left(\frac{y^{-1}}{p}\right) = \left(\frac{y}{p}\right)$ we have

$$\frac{1}{\sqrt{p-1}} \frac{1}{\sqrt{p}} \left[\sum_{z=0}^{p-1} \left(\frac{z}{p}\right) \omega_p^z \right] \sum_{y=1}^{p-1} \left(\frac{y}{p}\right) \omega_p^{-ys} |y\rangle$$

So we are left to evaluate $\sum_{z=0}^{p-1} \left(\frac{z}{p}\right) \omega_p^z$, which is the Gauß sum [3, 20], and is \sqrt{p} if $p \equiv 1 \pmod{4}$ and is $i\sqrt{p}$ if $p \equiv 3 \pmod{4}$. Hence, up to a global constant which we can ignore, the state follows. \square

Corollary 1 *Algorithm 1 can be used to solve the Shifted Jacobi Symbol Problem.*

Proof: We start with the uniform superposition of \mathbb{Z}_n and calculate the function value f_s for each element:

$$\frac{1}{\sqrt{n}} \sum_{x \in \mathbb{Z}_n} |x, 0\rangle \longrightarrow \frac{1}{\sqrt{n}} \sum_{x \in \mathbb{Z}_n} |x, \left(\frac{x+s}{n}\right)\rangle.$$

Next, we measure if the rightmost value is non-zero. If this is the case, which happens with probability $\phi(n)/n$ (where ϕ is Euler's phi function obeying $\phi(n) = |\mathbb{Z}_n^*|$), the state has collapsed to the superposition:

$$\frac{1}{\sqrt{\phi(n)}} \sum_{x \in \mathbb{Z}_n^*} |x, \left(\frac{x+s}{n}\right)\rangle.$$

Otherwise, we simply try again the same procedure. (The success probability $\phi(n)/n$ is lower bounded by $\Omega(1/\log(\log n))$, see [1], hence we can expect to be successful after $O(\log(\log n))$ trials.)

We continue with the reduced state by changing the phase of $|x\rangle$ to $(\frac{x+s}{n})$ and uncomputing the function value again, giving

$$\frac{1}{\sqrt{\phi(n)}} \sum_{x \in \mathbb{Z}_n} \left(\frac{x+s}{n} \right) |x\rangle.$$

Let $n = p_1 \cdot p_2 \cdots p_k$ be the prime decomposition of n such that $\mathbb{Z}_n = \mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_k}$. Using Shor's algorithm[18], we can determine these factors efficiently. Because $(\frac{x+s}{n}) = (\frac{x+s_1}{p_1}) \cdot (\frac{x+s_2}{p_2}) \cdots (\frac{x+s_k}{p_k})$, we can just consider each p_j component separately (with $s_1 \equiv s \bmod p_1$, $s_2 \equiv s \bmod p_2$, et cetera). Hence, by performing the 'inverse Chinese remainder' map $|x\rangle \longrightarrow |x \bmod p_1, \dots, x \bmod p_k\rangle$, we obtain the state

$$\sum_{x_1 \in \mathbb{Z}_{p_1}} \cdots \sum_{x_k \in \mathbb{Z}_{p_k}} \left(\frac{x_1 + s_1}{p_1} \right) \cdots \left(\frac{x_k + s_k}{p_k} \right) |x_1, \dots, x_k\rangle = \bigotimes_{j=1}^k \sum_{x_j \in \mathbb{Z}_{p_j}} \left(\frac{x_j + s_j}{p_j} \right) |x_j\rangle.$$

But now we use Algorithm 1 on each factor to get $|-s_1, \dots, -s_k\rangle$, after which the Chinese remainder theorem gives us the answer s . \square

We now give an algorithm for the above problem when also n is unknown. In addition to using known techniques, the algorithm depends on the fact that sampling the Fourier transform of the shifted Legendre Symbol results in the uniform distribution on \mathbb{Z}_n^* .

Algorithm 2 (Shifted Jacobi Symbol Problem, unknown n)

Input: An integer M and a function $f_s : \{0, \dots, M-1\} \rightarrow \{-1, 0, 1\}$ such that $f_s(x) = (\frac{x+s}{n})$ for some integer n , with $n^2 < M$

Output: n and s .

1. Create the following state as in Corollary 1:

$$c \cdot \sum_{x=0}^{M-1} \left(\frac{x+s}{n} \right) |x\rangle$$

2. Compute the Fourier transform over \mathbb{Z}_M .
3. Measure, with outcome i , and use continued fractions on i and M , returning j/n .
4. Run Algorithm 1 using f_s and n .

Theorem 2 Algorithm 1 solves the Shifted Jacobi Symbol Problem with unknown n in quantum polynomial time with high probability.

Proof: Let $|\psi_s\rangle = \frac{1}{\sqrt{\phi(n)}} \sum_{x=0}^{n-1} \left(\frac{x+s}{n}\right) |x\rangle$ be the state after the setup in Corollary 1 and let $|\tilde{\psi}_s\rangle = c \sum_{x=0}^{M-1} \left(\frac{x+s}{n}\right) |x\rangle$ be the repeated version in Algorithm 2, where c is the normalizing constant. We can relate the distributions induced by Fourier sampling $|\phi_s\rangle$ and $|\tilde{\phi}_s\rangle$ using the discussion in Section 2. If $M = n$ then Lemma 1 implies that i is uniformly distributed over \mathbb{Z}_n^* and we would be done since the denominator returned by continued fractions is n in this case. However this will still be the case even if $M \neq n$. If M is a multiple of n and if the Fourier transform of $|\psi_s\rangle$ is $\sum_{x=0}^{n-1} \alpha_x |x\rangle$, then the Fourier transform of $|\tilde{\psi}_s\rangle$ is $\sum_{x=0}^{n-1} \alpha_x |M/n \cdot x\rangle$, so we get what we want. If M is not a multiple but is large enough, the distributions as discussed in Section 2 are ϵ -close. \square

Lemma 1 *Let n be an odd square free integer. If we apply the quantum Fourier transform over \mathbb{Z}_n to the superposition of the states $\left(\frac{x+s}{n}\right) |x\rangle$ for all $x \in \mathbb{Z}_n$, we establish the evolution*

$$\frac{1}{\sqrt{\phi(n)}} \sum_{x \in \mathbb{Z}_n} \left(\frac{x+s}{n}\right) |x\rangle \longrightarrow \frac{i^{(n-1)^2/4}}{\sqrt{\phi(n)}} \sum_{y \in \mathbb{Z}_n} \omega_n^{-sy} \left(\frac{y}{n}\right) |y\rangle.$$

Proof: First, we note that we can rewrite the output as

$$\frac{1}{\sqrt{n \cdot \phi(n)}} \sum_{y \in \mathbb{Z}_n} \sum_{x \in \mathbb{Z}_n} \left(\frac{x+s}{n}\right) \omega_n^{xy} |y\rangle = \frac{1}{\sqrt{n \cdot \phi(n)}} \sum_{y \in \mathbb{Z}_n} \omega_n^{-sy} \left[\sum_{x \in \mathbb{Z}_n^*} \left(\frac{x}{n}\right) \omega_n^{xy} \right] |y\rangle,$$

by substituting x with $x+s$ in the summation and using the fact that $\left(\frac{x}{n}\right) = 0$ for all $x \notin \mathbb{Z}_n^*$.

The amplitudes between the square brackets depend on y in the following way. First, we consider the case when y is co-prime to n , that is: $y \in \mathbb{Z}_n^*$, and there exists also an inverse $y^{-1} \in \mathbb{Z}_n^*$. We then see that

$$\sum_{x \in \mathbb{Z}_n^*} \left(\frac{x}{n}\right) \omega_n^{xy} = \left(\frac{y^{-1}}{n}\right) \sum_{z \in \mathbb{Z}_n^*} \left(\frac{z}{n}\right) \omega_n^z,$$

where we used the substitution $x \leftarrow zy^{-1}$ and the multiplicativity of the Jacobi symbol.

Next, we look at the case where n and y have a common, non-trivial, factor f . We say that $n = mf$ and $y = rf$, and we know that f and m are co-prime (because n is square free). The Chinese remainder theorem tells us that there is a bijection between the elements $x \in \mathbb{Z}_n$ and the coordinates $(x \bmod m, x \bmod f) \in \mathbb{Z}_m \times \mathbb{Z}_f$, which also establishes a one-to-one mapping between \mathbb{Z}_n^* and $\mathbb{Z}_m^* \times \mathbb{Z}_f^*$. This allows us to rewrite the expression as follows.

$$\begin{aligned} \sum_{x \in \mathbb{Z}_n^*} \left(\frac{x}{n}\right) \omega_n^{xy} &= \sum_{x \in \mathbb{Z}_{mf}^*} \left(\frac{x}{mf}\right) \omega_{mf}^{xrf} \\ &= \sum_{x \in \mathbb{Z}_{mf}^*} \left(\frac{x \bmod m}{m}\right) \left(\frac{x \bmod f}{f}\right) \omega_m^{xr} \\ &= \sum_{x_1 \in \mathbb{Z}_m^*} \left(\frac{x_1}{m}\right) \omega_m^{x_1 r} \sum_{x_2 \in \mathbb{Z}_f^*} \left(\frac{x_2}{f}\right). \end{aligned}$$

Because f is odd and square free $\sum_{x \in \mathbb{Z}_f^*} \left(\frac{x}{f}\right) = 0$, and hence the above term equals zero. This concludes the proof of the lemma. \square

4 An Algorithm for General Finite Fields

Here we will solve the general case of the Shifted Legendre Symbol Problem for any finite field \mathbb{F}_q . From now on $q = p^r$, with p an odd prime and the degree r an integer. See Section 2 for details about finite fields. The idea used for the SLSP algorithm cannot be used directly here, since the matrix is no longer circulant. To get around that problem, we use the following map:

Lemma 2 (Trace-Fourier Transform over \mathbb{F}_q) *The unitary mapping*

$$|x\rangle \longrightarrow \frac{1}{\sqrt{q}} \sum_{y \in \mathbb{F}_q} \omega_p^{\text{Tr}(xy)} |y\rangle$$

is computable in polynomial time.

Proof: Assume that the mapping

$$|x\rangle \longrightarrow \bigotimes_{j=0}^{r-1} |\text{Tr}(xX^j)\rangle.$$

can be computed in polynomial time. First apply this map, and then compute the Fourier transform over \mathbb{Z}_p^r . This gives us the final state

$$\bigotimes_{j=0}^{r-1} \frac{1}{\sqrt{p}} \sum_{y_j \in \mathbb{F}_p} \omega_p^{\text{Tr}(xX^j)y_j} |y_j\rangle = \frac{1}{\sqrt{q}} \sum_{y \in \mathbb{F}_q} \omega_p^{\text{Tr}(xy)} |y\rangle.$$

We will now show that the map

$$|x\rangle \longrightarrow |\text{Tr}(x), \text{Tr}(xX), \dots, \text{Tr}(xX^{r-1})\rangle$$

is reversible. Let $T(x) = [\text{Tr}(x), \text{Tr}(xX), \dots, \text{Tr}(xX^{r-1})]$. T is a linear functional since Tr is, so if $T(a) = T(b)$ then $T(a - b)$ is the zero vector. We will show that $T(x)$ is not the zero vector except for $x = 0$. Suppose $T(x)$ is the zero vector. Since Tr is not the zero map, choose $a \in \mathbb{F}_q$ such that $\text{Tr}(a) \neq 0$. Choose z_0, \dots, z_{r-1} such that $\sum_j z_j xX^j = a$. Then $\text{Tr}(a) = \text{Tr}(\sum_j z_j xX^j) = \sum_j z_j \text{Tr}(xX^j) = 0$, since $\text{Tr}(xX^j) = 0$ for all j . But this is a contradiction by the choice of a . So T is one-to-one.

We will now show that the map is computable in polynomial time. It is enough if x can be computed from $\text{Tr}(x), \text{Tr}(xX), \dots, \text{Tr}(xX^{r-1})$. But the equations $\text{Tr}(x) = \sum_{j=0}^{r-1} x_j \text{Tr}(X^j)$, $\text{Tr}(xX) = \sum_{j=0}^{r-1} x_j \text{Tr}(X^{j+1})$, \dots , $\text{Tr}(xX^{r-1}) = \sum_{j=0}^{r-1} x_j \text{Tr}(X^{j+r-1})$ are r linear equations in r unknowns, and the values $\text{Tr}(x), \text{Tr}(xX), \dots, \text{Tr}(xX^{r-1})$ and $\text{Tr}(X^j)$ for all j are known, so the coefficients x_j of x can be solved for using linear algebra. \square

(We recently learned that, independently, de Beaudrap *et al.* [2] have used a transform closely related to the above for the construction of a different quantum algorithm.)

Theorem 3 *Algorithm 3 (see below) solves the Shifted Quadratic Character Problem over any finite field with two queries and in polynomial time with probability exponentially close to one.*

The presentation of the algorithm below differs from the SLSP algorithm in that it does not use approximations. Approximations work here also, but here we show how the problem can be solved exactly if the base field is of fixed size (so that the Fourier transform is not approximate). Also notice that the Trace-Fourier transform is not a unique solution to this problem, any linear functional will work in place of Tr .

Algorithm 3

Input: A power of a prime $q = p^r$ and a function f_s such that $f_s(x) = \chi(x + s)$.

Output: s .

1. Use the Fourier transform over \mathbb{Z}_{q+1} on $|0\rangle$, and two queries to f_s to create (with probability $q/(q+1)$) the state

$$\frac{1}{\sqrt{q}} \sum_{x \in \mathbb{F}_q} \chi(x + s) |x\rangle + \frac{1}{\sqrt{q}} |\delta\rangle.$$

With probability $1/(q+1)$ this step gives s directly.

2. Compute the Trace-Fourier transform of Lemma 2 over \mathbb{F}_q . In the proof it will be shown that the output of this transform equals

$$\frac{1}{q} \left[\sum_{z \in \mathbb{F}_q} \chi(z) \omega_p^{\text{Tr}(z)} \right] \left(\sum_{y \in \mathbb{F}_q} \chi(y) \omega_p^{\text{Tr}(-sy)} |y\rangle \right) + \frac{1}{\sqrt{q}} |\delta\rangle. \quad (1)$$

The term between the square brackets is known the quadratic Gauß sum $G(\mathbb{F}_q)$, with $G(\mathbb{F}_q) = (-1)^{r-1} i^{r(p-1)^2/4} \sqrt{q}$ (see Theorem 11.5.4 in [3]). With this knowledge we can perform the next step.

3. Uncompute the phases $\chi(y)$ for $y \neq 0$, and change the dummy vector to $(-1)^{r-1} i^{r(p-1)^2/4} |0\rangle$. By dropping the general phase $G(\mathbb{F}_q)/\sqrt{q}$, we can now write

$$\frac{1}{\sqrt{q}} \sum_{y \in \mathbb{F}_q} \omega_p^{\text{Tr}(-sy)} |y\rangle$$

for the state.

4. Finally, compute the inverse Trace-Fourier transform over \mathbb{F}_q . This gives us the requested shift parameter as $|-s\rangle$.

Proof: For the first step we create, with one call to f_s the superposition

$$\frac{1}{\sqrt{q+1}} \sum_{x \in \mathbb{F}_q} |x, f_s(x)\rangle + \frac{1}{\sqrt{q+1}} |\delta, 1\rangle,$$

where δ denotes a ‘dummy state’. Next, we measure if the rightmost bit is zero. If this is the case (probability $1/(q+1)$), the state has collapsed to $|-s, 0\rangle$, which tells us the value s immediately. Otherwise, we are left with the superposition of the entries x with $f_s(x) = \pm 1$ and the dummy

state. This enables us to create the proper phases $f_s(x) = \chi(x + s)$ and uncompute (with a second f_s query) the rightmost bit, which we will ignore from now on.

At step 2, we perform the Trace-Fourier transform to the state, yielding

$$\frac{1}{q} \sum_{x \in \mathbb{F}_q} \sum_{y \in \mathbb{F}_q} \chi(x + s) \omega_p^{\text{Tr}(xy)} |y\rangle + \frac{1}{\sqrt{q}} |\delta\rangle.$$

We rewrite this expression as follows: replace x with $z = xy + sy$, and use the multiplicativity of χ and the linearity of the trace in $\chi(zy^{-1}) \omega_p^{\text{Tr}(z-sy)} = \chi(z) \omega_p^{\text{Tr}(z)} \cdot \chi(y) \omega_p^{\text{Tr}(-sy)}$. This proves the validity of Equation 1. \square

5 Conclusion and Open Problems

We have shown the existence of efficient quantum algorithms for several versions of the ‘Shifted Quadratic Character Problem’. The classical complexity of these problems remains open. In the light of Shor’s result[18], we would also like to know whether the problems become classically tractable if we assume that factoring is easy.

6 Acknowledgements

WvD was supported by the Institute for Logic, Language and Computation in Amsterdam, the EU fifth framework project QAIP IST-1999-11234, and the TALENT grant S 62-552 of the Netherlands Organization for Scientific Research (NWO).

References

- [1] Eric Bach and Jeffrey Shallit, *Algorithmic Number Theory, Volume 1: Efficient Algorithms*, MIT Press (1996)
- [2] J. Niel de Beaudrap, Richard Cleve, John Watrous, “Quantum Fourier transforms for extracting hidden linear structures in finite fields”, quant-ph archive no. 0011065 (2000)
- [3] Bruce C. Berndt, Ronald J. Evans, and Kenneth S. Williams, *Gauss and Jacobi Sums*, John Wiley & Sons (1998)
- [4] Ethan Bernstein and Umesh Vazirani, “Quantum Complexity Theory”, *SIAM Journal on Computing*, Volume 26(5), pp. 1411–1473 (1997)
- [5] Dan Boneh and Richard J. Lipton, “Quantum Cryptanalysis of Hidden Linear Functions (Extended Abstract)” *Advances in Cryptology—CRYPTO’95*, pp. 424–437 (1995)
- [6] Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca, “Quantum algorithms revisited”, *Proceedings of the Royal Society of London A*, Volume 454, pp. 339–354 (1998)
- [7] Henri Cohen, *A Course in Computational Algebraic Number Theory*, Graduate Texts in Mathematics 138, Springer-Verlag (1993)

- [8] Wim van Dam, “Quantum Algorithms for Weighing Matrices and Quadratic Residues”, quant-ph archive no. 0008059 (2000)
- [9] Ivan B. Damgård, “On the randomness of Legendre and Jacobi sequences”, *Advances in Cryptology—Proceedings of CRYPTO’88*, pp. 163–172 (1990)
- [10] Mark Ettinger and Peter Høyer, “On quantum algorithms for noncommutative hidden subgroups”, *16th Annual Symposium on Theoretical Aspects in Computer Science*, Lecture Notes in Computer Science, Volume 1563, pp. 478–487 (1999)
- [11] Michaelangelo Grigni, Leonard Schulman, and Umesh Vazirani, “Quantum Mechanical Algorithms for the Non-Abelian Hidden Subgroup Problem”, Manuscript (1997)
- [12] Lisa Hales and Sean Hallgren, “An Improved Quantum Fourier Transform Algorithm and Applications”, *Proceedings of the 41st Annual Symposium on Foundations of Computer Science* (2000)
- [13] Sean Hallgren and Alexander Russell and Amnon Ta-Shma, “Normal Subgroup Reconstruction and Quantum Computation Using Group Representations”, *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pp. 627–635 (2000)
- [14] Alexey Yu. Kitaev, “Quantum measurements and the Abelian stabilizer problem”, quant-ph report no. 9511026; ECCC Report TR96-003 (1995)
- [15] Rudolph Lidl and Harald Niederreiter, *Finite Fields*, Cambridge University Press, 2nd edition (1997)
- [16] Michele Mosca and Artur Ekert, “The Hidden Subgroup Problem and Eigenvalue Estimation on a Quantum Computer”, *Proceedings of the 1st NASA International Conference on Quantum Computing and Quantum Communications*, Lecture Notes in Computer Science, Volume 1509 (Editor C.P. Williams), pp. 174–188 (1999)
- [17] René Peralta, “On the distribution of quadratic residues and nonresidues modulo a prime number”, *Mathematics of Computation*, Volume 58, pp. 433–440 (1992)
- [18] Peter W. Shor, “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer”, *SIAM Journal on Computing*, Volume 26(5), pp. 1484–1509 (1997)
- [19] Daniel R. Simon, “On the power of quantum computation”, *SIAM Journal on Computing*, Volume 26(5), pp. 1474–1483 (1997)
- [20] Audrey Terras, *Fourier Analysis on Finite Groups and Applications*, London Mathematical Society Student Texts 43, Cambridge University Press (1999)

A From Legendre Symbols to Gauß Sums

A.1 Legendre Symbol, Jacobi Symbol, and Quadratic Character

In this appendix, p is a prime, q is the prime power p^r with degree r , and n is a (typically non-prime) positive integer. We denote the field of size p by \mathbb{Z}_p , instead of the perhaps more accurate $\mathbb{Z}/(p\mathbb{Z})$. Similarly, the ring induced by mod n addition and multiplication is \mathbb{Z}_n . The finite field of size q is described by \mathbb{F}_q , and hence $\mathbb{F}_p = \mathbb{Z}_p$, but $\mathbb{F}_{p^2} \neq \mathbb{Z}_{p^2}$. The respective multiplicative subgroups are indicated by the $*$ superscript: \mathbb{Z}_p^* , \mathbb{Z}_n^* and \mathbb{F}_q^* .

The *Legendre symbol* indicates if a non-zero element is a square modulo p or not:

$$\left(\frac{x}{p}\right) = \begin{cases} 0 & \text{if } x \equiv 0 \pmod{p} \\ +1 & \text{if there exists a } y \neq 0 \text{ such that } y^2 \equiv x \pmod{p} \\ -1 & \text{if for all } y: y^2 \not\equiv x \pmod{p}. \end{cases}$$

Let $n = p_1 \cdot p_2 \cdots p_k$ be the prime factor decomposition of n . The *Jacobi symbol* generalizes the Legendre symbol for all rings \mathbb{Z}_n in the following way:

$$\left(\frac{x}{n}\right) = \left(\frac{x}{p_1}\right) \cdot \left(\frac{x}{p_2}\right) \cdots \left(\frac{x}{p_k}\right).$$

Clearly, $\left(\frac{x}{n}\right) = 0$ for all x not co-prime to n . Note that $\left(\frac{x}{n}\right) = +1$ does not always imply that there is a y with $y^2 \equiv x \pmod{n}$. Take, for example, $\left(\frac{2}{9}\right) = 1$.

For finite fields \mathbb{F}_q , the Legendre symbol becomes the *quadratic character* χ , which is defined

$$\chi(x) = \begin{cases} 0 & \text{if } x = 0 \\ +1 & \text{if there exists a } y \neq 0 \text{ such that } y^2 = x \\ -1 & \text{if for all } y: y^2 \neq x. \end{cases}$$

for all $x \in \mathbb{F}_q$.

A.2 Basic Properties

The Legendre symbol, Jacobi symbol and quadratic character are all three *multiplicative characters* because they obey $(xy/p) = (x/p)(y/p)$, $(xy/n) = (x/n)(y/n)$, and $\chi(xy) = \chi(x)\chi(y)$, respectively. This implies a series of results.

Let g be a generator of \mathbb{Z}_p^* . Because the multiplicative subgroup has $p-1$ elements, we know that $g^i = g^j \pmod{p}$ if and only if $i = j \pmod{p-1}$. Hence, the quadratic equation $(g^j)^2 = g^{2j} = g^i \pmod{p}$ is correct if and only if $2j = i \pmod{p-1}$.

For an odd prime p , there can only exist a j with $(g^j)^2 = g^i \pmod{p}$ when i is even, as $p-1$ is even. Obviously, if i is even, then g^j with $j = \frac{i}{2}$ gives also a solution. In short: $(g^i/p) = (-1)^i$. This proves that $\frac{p-1}{2}$ of the elements x of \mathbb{Z}_p^* are a quadratic residue with $(x/p) = +1$, while the other $\frac{p-1}{2}$ are non-squares.

If p is even, then for all i either i or $(i+p-1)$ will be even. Hence, either $j = \frac{i}{2}$ or $j = \frac{i+p-1}{2}$ gives a proper solution to the equality $(g^j)^2 = g^i \pmod{p}$. This proves that all elements $x \in \mathbb{Z}_p^*$ are quadratic residues with $(x/p) = +1$. This rather redundant proof for the only existing case $p = 2$ is justified by the following lemma.

Lemma 3 For any finite field \mathbb{F}_{p^r} , we have for the summation of its quadratic character values

$$\sum_{x \in \mathbb{F}_{p^r}} \chi(x) = \begin{cases} 0 & \text{if } p \text{ is odd} \\ p^r - 1 & \text{if } p \text{ is even.} \end{cases}$$

Proof: Every multiplicative group $\mathbb{F}_{p^r}^*$ has a generator g with period $p^r - 1$. Use this in combination with the proof method of the preceding paragraphs. \square

We will reach a similar result for the summation of the Jacobi symbol values over \mathbb{Z}_n , when n is odd and squarefree. Let again $n = p_1 \cdots p_k$ be the prime decomposition. The Chinese remainder theorem tells us that the mapping $x \in \mathbb{Z}_n \rightarrow (x \bmod p_1, \dots, x \bmod p_k) \in \mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_k}$ is a bijection. (All p_i terms are different, because we assumed n to be square free.) This enables us to prove the following lemma.

Lemma 4 Let n be an odd, square free integer. The summation of all the Jacobi values of \mathbb{Z}_n obeys

$$\sum_{x \in \mathbb{Z}_n} \left(\frac{x}{n} \right) = 0.$$

Proof: Let $n = p_1 \cdots p_k$ be the decomposition of n into its prime factors. The definition of the Jacobi symbol in combination with Chinese remainder theorem yields the equality

$$\begin{aligned} \sum_{x \in \mathbb{Z}_n} \left(\frac{x}{n} \right) &= \sum_{x \in \mathbb{Z}_n} \left(\frac{x}{p_1} \right) \cdots \left(\frac{x}{p_k} \right) \\ &= \sum_{x_1 \in \mathbb{Z}_{p_1}} \cdots \sum_{x_k \in \mathbb{Z}_{p_k}} \left(\frac{x_1}{p_1} \right) \cdots \left(\frac{x_k}{p_k} \right) \\ &= \left(\sum_{x_1 \in \mathbb{Z}_{p_1}} \left(\frac{x_1}{p_1} \right) \right) \cdots \left(\sum_{x_k \in \mathbb{Z}_{p_k}} \left(\frac{x_k}{p_k} \right) \right). \end{aligned}$$

By the previous lemma we know that each $\sum_{x \in \mathbb{Z}_p} (x/p)$ is zero, hence the above product is zero as well. \square

A.3 Gauß Sums

Let ω_p denote the complex root $e^{2\pi i/p}$. The *trace* of an element $x \in \mathbb{F}_{p^r}$ is defined by $\text{Tr}(x) = \sum_{j=0}^{r-1} x^{p^j}$. It can be shown that for every $x \in \mathbb{F}_{p^r}$, its trace is an element of the base-field: $\text{Tr}(x) \in \mathbb{F}_p$. When we write $\omega_p^{\text{Tr}(x)}$ we interpret the value $\text{Tr}(x)$ as an element of the set $\{0, 1, \dots, p-1\} \subset \mathbb{Z}$.

Definition 5 For the field \mathbb{Z}_p , the ring \mathbb{Z}_n and the finite field \mathbb{F}_{p^r} we define the quadratic Gauß

sum G by

$$\begin{aligned} G(\mathbb{Z}_p) &= \sum_{x \in \mathbb{Z}_p} \left(\frac{x}{p} \right) \omega_p^x, \\ G(\mathbb{Z}_n) &= \sum_{x \in \mathbb{Z}_n} \left(\frac{x}{n} \right) \omega_n^x, \\ G(\mathbb{F}_{p^r}) &= \sum_{x \in \mathbb{F}_{p^r}} \chi(x) \omega_p^{\text{Tr}(x)}. \end{aligned}$$

It is not immediately clear that this definition does not give contradicting values for the identical cases $G(\mathbb{Z}_p)$ and $G(\mathbb{F}_p)$. However, the next result shows that this conflict does not occur. We will not give the proofs of the following lemma as that goes far beyond the scope of this article. Instead, the curious reader is referred to the book by Berndt *et al.*[3]

Lemma 5 *Let p be an odd prime and n an odd square free integer. The following equalities hold for the different quadratic Gauß sums:*

$$\begin{aligned} G(\mathbb{Z}_p) &= \begin{cases} \sqrt{p} & \text{if } p \equiv 1 \pmod{4} \\ i\sqrt{p} & \text{if } p \equiv 3 \pmod{4} \end{cases} \\ G(\mathbb{Z}_n) &= \begin{cases} \sqrt{n} & \text{if } n \equiv 1 \pmod{4} \\ i\sqrt{n} & \text{if } n \equiv 3 \pmod{4} \end{cases} \\ G(\mathbb{F}_{p^r}) &= \begin{cases} -\sqrt{p^r} & \text{if } p \equiv 1 \pmod{4} \text{ and } r \text{ is even} \\ \sqrt{p^r} & \text{if } p \equiv 1 \pmod{4} \text{ and } r \text{ is odd} \\ -\sqrt{p^r} & \text{if } p \equiv 3 \pmod{4} \text{ and } r \equiv 0 \pmod{4} \\ i\sqrt{p^r} & \text{if } p \equiv 3 \pmod{4} \text{ and } r \equiv 1 \pmod{4} \\ \sqrt{p^r} & \text{if } p \equiv 3 \pmod{4} \text{ and } r \equiv 2 \pmod{4} \\ -i\sqrt{p^r} & \text{if } p \equiv 3 \pmod{4} \text{ and } r \equiv 3 \pmod{4} \end{cases} \end{aligned}$$

Note that indeed $G(\mathbb{Z}_p) = G(\mathbb{F}_p)$.